Linguistic Modelling Using a Semi-Naïve Bayes Framework

N. J. Randon and **J. Lawry**

A.I. Group,
Department of Engineering Mathematics
University of Bristol BS8 1TR
United Kingdom
{Nick.Randon, J.Lawry}@bris.ac.uk

Abstract

A random set semantics is presented as a knowledge representation framework for learning linguistic prototypes. Within this framework a number of algorithms for learning prototypes are presented, based on grouping certain sets of attributes and evaluating joint mass assignments on labels. Such prototypes are then combined with a semi-Naïve Bayes classifier in order to determine classification probabilities. The potential of such linguistic classifiers is then illustrated by their application to a number of toy and benchmark problems.

Keywords: Random Sets, Mass Assignment, Label Prototypes, Semi-Naïve Bayes.

1 Introduction

The concept of "computing with words" in fuzzy logic was introduced by Zadeh [15]. He stated that fuzzy logic offered an intuitive method for modelling natural language where the meaning of words such as "small, medium and large" could be represented by fuzzy sets. This is a particularly appealing concept when dealing with real word problems where there is often imprecision and ambiguity. Zadeh's idea was centred on using linguistic variables to represent linguistic constraints. This however generates a numbers of problems relating to both the semantics and computational complexity (see [12] for a discussion of these issues)

Here we present an extension to an alternative method proposed by Lawry [11]. The method uses random sets as a way of choosing appropriate labels for a given variable. Further more it builds on the ability of fuzzy sets to partition a domain into linguistic descriptions coupled with a new label semantics to give a feasible and consistent linguistic inference mechanism (see [8] and [10]). The method uses mass assignments on labels to provide a measure of the appropriateness of a given label for a particular value. Classification is performed using prototypes, consisting of a vector of mass assignments on labels, giving the appropriateness of words as labels for the feature values for a certain class. These are then used in conjunction with a Naïve or Semi-Naïve Bayes classifier (see [6] and [7]). As such the prototypes provide an aggregated linguistic description of the examples of that class in the database.

This paper introduces techniques that can be applied to counter the "the curse of dimensionality"[4]. It also provides a method for solving non-decomposable problems such as XOR, by introducing Semi-Naïve Bayes to weaken the independence assumption of Naïve Bayes [7].

2 Label Semantics

Consider an attribute describing a universe of discourse Ω, assumed in this context to be a closed interval of real numbers. A finite set of labels LA is defined over Ω to form a linguistic covering. For example, in the classification of diabetes the diastolic blood pressure may be recorded. From reading this data a doctor may conjecture, "*The measured blood pressure is very high*". This would mean that an appropriate label for describing blood pressure would be *very high* $\in LA$. More formally, for each label $l \in LA$ a fuzzy set μ_l is defined, representing it's meaning. Any value x then generates a mass assignment (see Baldwin [1]) on labels as follows:

Let $\{l_1,\ldots,l_n\} = \{l \in LA \mid \mu_l(x) > 0\}$ be ordered such that $\mu_l(x) \geq \mu_{l+1}(x)$ then the mass assignment generated by x is:

$$\{l_1,\ldots,l_i\} : \mu_{l_i}(x) - \mu_{l_{i+1}}(x) \text{ For } i = 1,\ldots,n-1$$

$$\{l_1,\ldots,l_n\} : \mu_{l_n}(x)$$

$$\phi : 1 - \mu_{l_1}(x)$$

It is assumed that the distribution is on a random set $\mathcal{D}_x$ describing the set of labels deemed as appropriate for x, as it varies across some population of voters and is denoted $m_{\mathcal{D}_x}$. (See [11] and [12])

In practice, it is undesirable to have mass associated with the empty set; hence the further assumption is made that $\forall x \in \Omega \max_{l \in LA} \mu_l(x) = 1$. Here this is accomplished using trapezoidal fuzzy sets with a 50% overlap, as in figure 1:

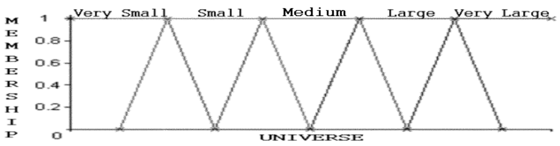

Figure 1: Trapezoidal Fuzzy Sets With 50% Overlap

3 Label Prototypes for Modelling Classification Problems

Consider a classification problem where a prototype is generated from a set of attributes $X_1,\ldots,X_n$, describing the classes $C_1,\ldots,C_k$,. In this case a finite set of labels LA_j is defined over each of the variables X_j. The database is partitioned into subsets corresponding to each class as follows:

Consider a training set of examples $DB = \{\langle x_1(i),\ldots,x_n(i)\rangle \mid i = 1,\ldots,N\}$ where each example i has associated class $C(i)$. From this database a set of sub-databases can be obtained for each class, $DB_j = \{\langle x_1(i),\ldots,x_n(i)\rangle \mid C(i) = C_j\}$.

The attributes $X_1,\ldots,X_n$ are now partitioned into subsets $S_1,\ldots,S_w$ where $w \le n$ and for each S_i a joint mass assignment $m_{i,j}$ is determined as follows: Suppose, w.l.o.g. $S_i = \{x_1,\ldots,x_v\}$ then the joint mass assignment is:

$$\forall T_1 \times \ldots \times T_v \in 2^{LA_1} \times \ldots \times 2^{LA_v}$$

$$m_{i,j}(T_1,\ldots,T_v) = \frac{1}{|DB_j|} \sum_{k \in DB_j} \prod_{r=1}^{v} m_{\mathcal{D}_{X_{r(k)}}}(T_r)$$

Section five describes how to automatically learn groupings of attributes for each class.

4 Estimating Classification Probabilities From Prototypes

We now give details of how to estimate class probabilities using label prototypes, which then can be incorporated into a Bayesian framework.

In machine learning it is common to make the "*Naïve Bayes assumption*" (see [7]) that all variables are conditionally independent given a class. This assumption is weakened in our case as the prototype describing a class may contain joint mass assignments; hence the classifier described is based on Semi-Naïve Bayes [6].

Bayes theorem states that for a vector of attribute values $\langle x_1,\ldots,x_n\rangle$, the class probability can be expressed as follows:

$$\Pr(C_j \mid x_1,\ldots,x_n) = \frac{p(x_1,\ldots,x_n \mid C_j)\Pr(C_j)}{p(x_1,\ldots,x_n)}$$

This can be simplified to the following estimate, for the purpose of classification:

$$\Pr(C_j \mid x_1,\ldots,x_n) \propto p(x_1,\ldots,x_n \mid C_j)|C_j|$$

There now remains the problem of how to estimate the density function $p(x_1,\ldots,x_n|C_j)$. In the current context it is assumed that the prototype for each class can be used to estimate the density function, as follows:

Consider the joint mass assignment generated for the attribute grouping S_i given class C_j. Then if we assume that there is a uniform prior distribution on $\times_{r=1}^{v} \Omega_r$ then the prior mass assignment on $\times_{r=1}^{v} 2^{LA_r}$ is given by:

$$m(T_1,\ldots,T_v) = \int_{\Omega_1,\ldots,\Omega_v} m_{\mathcal{D}_{X_1}}(T_1) \times \ldots \times m_{\mathcal{D}_{X_v}}(T_v) \times u(x_1,\ldots,x_v)dx_1,\ldots,dx_v$$

$$= \prod_{r=1}^{v} \int_{\Omega_r} m_{\mathcal{D}_{X_r}}(T_r)u(x_r)dx_r$$

Where $u(x_1,\ldots,x_v)$ is the uniform distribution on $\times_{r=1}^{v} \Omega_r$ and $u(x_r)$ the uniform distribution on Ω_r. From this the density function for S_i based on $m_{i,j}$ is given by:

$$p(S_i \mid m_{i,j}) = p(x_1,\ldots,x_v \mid m_{i,j}) =$$

$$u(x_1,\ldots,x_v) \sum_{T_1 \times \ldots \times T_v} \frac{m_{i,j}(T_1,\ldots,T_v)}{m(T_1,\ldots,T_v)} \prod_{r=1}^{v} m_{\mathcal{D}_{X_r}}(T_r)$$

Hence taking $p(x_1,\ldots,x_v|C_j) \cong p(x_1,\ldots,x_v|m_{i,j})$ an estimate of the class probability is:

$$\Pr(C_j \mid x_1,\ldots,x_n) \propto |C_j| \prod_{r=1}^{w} p(S_r \mid m_{r,j})$$

5 Grouping Methods

We now consider methods for finding attribute groupings that increase discrimination in the model. For a given problem it is impractical to search the complete space of all attributes groupings and then partition to see if discrimination can be increased, as the search space would be exponential. For example, 20 attributes would generate a search of order 20! comparisons for a search limited to attribute groupings of two variables. Instead a heuristic search strategy is adopted. It is proposed that the search be guided by a measure of importance for each S_i, defined as follows:

5.1 Definition (Importance Measure)

Let the joint mass assignment for S_i given C_j be denoted $m_{i,j}$. For any input vector S_i the probability of class C_j can be estimated using Bayes theorem where:

$$\Pr(C_j \mid S_i) = \frac{p(S_i \mid m_{i,j})|C_j|}{p(S_i \mid m_{i,j})|C_j| + p(S_i \mid m_{i,\neg j})|\neg C_j|}$$

Hence $m_{i,\neg j}$ denotes the mass assignment for S_i given $\neg C_j$. The importance measured of group S_i for class C_j is then defined by:

$$IM_j(S_i) = \frac{\sum_{k \in DBj} \Pr(C_j \mid S_i(k))}{\sum_{k \in DB} \Pr(C_j \mid S_i(k))}$$

Careful limits must be set on the maximal size for groupings when running this algorithm, since as dimensionality increases the number of data points per focal element of the joint mass assignment decreases exponentially. The use of fuzzy sets in this context allows us to partially overcome this problem by trading off granularity against dimensionality and vice-versa. Two search strategies have been developed based on this measure.

5.2 Guided Breadth First Search

Consider a breadth first search where the most important current grouping S_i is combined with all the other current groupings to see if the combination significantly increases discrimination. Next the second most important unused grouping is tested with the remaining unused groupings and so on. At the next stage the new groupings produced are tested in a similar manner and this continues until a terminating condition is satisfied. This method provides a fairly extensive search of the space of the partitions, but does limit the structure of the groupings generated.

5.3 Guided Depth First Search

Alternatively, consider a depth first search where the most important grouping S_i is tested with all other groupings to see if the combination increases discrimination. Next any new grouping produced is tested with the unused groupings to see if discrimination is further increased. This continues until some termination condition is satisfied. Next the process is repeated with the next most important unused grouping and so on, until all unused grouping have been tested. This allows for a richer structure of groupings but has the disadvantage that some important groupings may be missed.

We now consider two ways of determining whether a pair of attribute groupings should be combined. The first is based on a direct measure of correlation and the second on a measure of the change in importance resulting from the grouping. Before we can define the above mentioned correlation measure we must first define what is meant by the focal sets for a mass assignment $m_{\mathcal{D}_x}$.

5.4 Definition (Focal Sets)

Let the focal sets, F, of $m_{\mathcal{D}_x}$ be given by:

$$F = \{S \subseteq LA \mid \exists x \in \Omega, m_{\mathcal{D}_x}(S) > 0\}$$

5.5 Definition (Correlation Measure)

Let F_1 be the focal sets for S_1 and F_2 the focal sets for F_2. Now let $m_{1,2,j}$ be the joint mass of $S_1 \cup S_2$ given C_j.

$$CORR(S_1, S_2) =$$

$$\sqrt{\frac{1}{|F_1||F_2|} \sum_{R \subseteq F_1} \sum_{T \subseteq F_2} \left(m_{1,2,j}(R,T) - m_{1,j}(R) m_{2,j}(T)\right)^2}$$

Here a threshold must be used to determine whether attributes should be grouped. The nearer the correlation measure gets to 1 the higher the correlation between attribute groups.

An alternative to measuring correlation is to use the importance measure, as a guide to whether attribute groups should be combined, by trying to maximise the importance of any new grouping formed.

5.6 Definition (Improvement Measure)

Suppose we have two subsets of attributes S_1 and S_2 then the improvement in importance obtained by combining them can be calculated as follows:

$$IPM_j(S_1,S_2) = \frac{\min(IM_j(S_1), IM_j(S_2))}{IM\ (S_1,S_2)}$$

Like the correlation measure a threshold is required, and in this instance the closer the improvement measure is to 0 the more likely that the attribute groups will be combined.

6 Performance On Model and Benchmark Problems

In this section we present a number of examples showing how the methods described in section 5 perform on model and real world problems.

6.1 Example (Non-Decomposable Model Problem)

In this example a figure of eight shape is generated according to the parametric equations $x = 2^{-0.5}(\sin 2t - \sin t)$ and $y = 2^{-0.5}(\sin 2t + \sin t)$ where $t \in [0,2\pi]$ as is illustrated in figure 2.

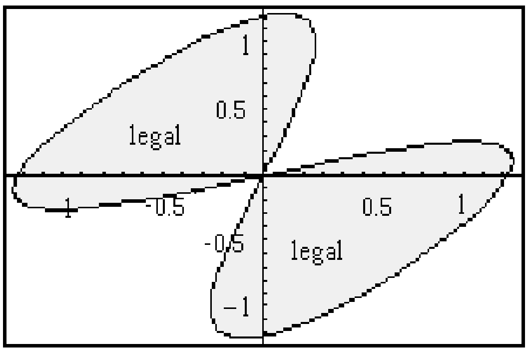

Figure 2: Figure Eight Classification Problem

A point on the $[-1.6,1.6]^2$ domain is classified as legal if it is contained within the figure and illegal if contained outside. The database contained 961 training examples of the X and Y co-ordinates and their associated class.

The prototypes obtained were generated from placing 5 labels over both attributes universes the meanings of which corresponded to uniformly distributed trapezoidal fuzzy sets. As there are only two attributes in this problem the choice of search method is arbitrary, as both will obtain the same results. For the correlation method a threshold of 0.005 was set, producing the following attribute groupings: *Legal* = $\{x, y\}$, *Illegal* = $\{x\}\{y\}$. For the improvement measure a threshold of 0.895 was set and the following groupings were generated: *Legal* = $\{x, y\}$, *Illegal* = $\{x, y\}$

Caution must be taken here as the thresholds used are not optimised for the problem and could suggest why the correlation method chooses slightly different grouping to the improvement measure. We can, however intuitively see why these groupings were chosen. If this problem is thought of as an XOR problem then clearly only a grouping of $\{x, y\}$ for the legal class is required for adequate classification.

From the groupings obtained it is possible to plot the posterior distributions learned from the data. Figure 3 shows these for the improvement measure approach. These suggest an inverse relationship between the legal and illegal distributions as would be expected.

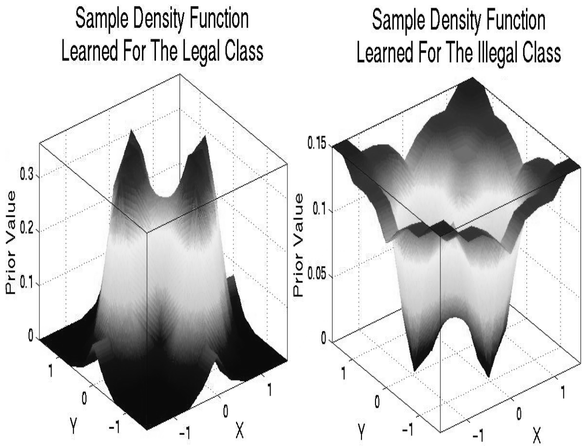

Figure 3: The Posterior Distributions Learned

The classifiers were then tested on a test set of 2119 unseen examples using the distributions for both the grouping methods and the following classification results were obtained:

Correlation Measure:

	Predicted Accuracy
Test	94.61248
Training	94.58897

Improvement Measure:

	Predicted Accuracy
Test	95.93573
Training	96.46202

Here slightly better results are obtained by using the improvement measure against the correlation method. It should be noted that if the Naïve Bayes independence assumption is used to classify then substantially poorer classification accuracy is obtained:

	Predicted Accuracy
Test	85.0662
Training	84.5994

6.2 Example (Sonar Data)

This problem is taken from the UCI online repository [14] and contains 208 examples obtained by bouncing sonar signals off metal cylinders and rocks. Each of the patters contains a set of 60 numbers in the range [0,1], which represent the energy within a particular frequency ban, over a certain time period. The integration aperture for high frequency occurs later in time, since these frequencies are transmitted later during the cipher.

It should be noted that the data used is for the "*aspect-angle dependent experiment*". Here the data set is split into training and test sets of 104 examples, where this split takes into account the aspect angle (see [9]). The data set was also normalised so that all attributes shared the same mean and standard deviation. Each attribute in the database had 2 labels placed over their domains in a non-uniform manner. This is performed by using a percentile method to obtain a crisp partition with an equal number of data point falling within each crisp set and then projecting trapezoidal fuzzy sets over this partition.

The depth and bread first search methods were applied with the two combination techniques and the following result were obtained:

Breath First Search Results:

Table 1: Correlation Measure Results

	Predicted Accuracy
Test	76.92308
Training	98.07692

Threshold = 0.005, **Max Grouping** = 4

Test Confusion Tableau:

Predicted Class/True Class	Rock	Metallic Cylinder
Rock	80.95238	19.04762
Metallic Cylinder	25.80645	74.19355

Training Confusion Tableau:

Predicted Class/ True Class	Rock	Metallic Cylinder
Rock	98.18182	1.81818
Metallic Cylinder	2.04082	97.95918

Table 2: Improvement Measure Results

	Predicted Accuracy
Test	90.38462
Training	99.03846

Threshold = 0.895, **Max Grouping** = 4

Tests Confusion Tableau:

Predicted Class/True Class	Rock	Metallic Cylinder
Rock	95.2381	4.7619
Metallic Cylinder	12.90323	87.09677

Training Confusion Tableau:

Predicted Class/True Class	Rock	Metallic Cylinder
Rock	98.18182	1.81818
Metallic Cylinder	0	100

Depth First Search Results

Table 3: Correlation Measure Results

	Predicted Accuracy
Test	91.34615
Training	97.11538

Threshold = 0.005, **Max Grouping** = 4

Tests Confusion Tableau:

Predicted Class/True Class	Rock	Metallic Cylinder
Rock	88.09524	11.90476
Metallic Cylinder	6.45161	93.54839

Training Confusion Tableau:

Predicted Class/True Class	Rock	Metallic Cylinder
Rock	98.18182	1.81818
Metallic Cylinder	4.08163	95.91837

Table 4: Improvement Measure Results

	Predicted Accuracy
Test	93.26923
Training	99.03846

Threshold = 0.895, **Max Grouping** = 4

Test Confusion Tableau:

Predicted Class/True Class	Rock	Metallic Cylinder
Rock	97.61905	2.38095
Metallic Cylinder	9.67742	90.32258

Training Confusion Tableau:

Predicted Class/ True Class	Rock	Metallic Cylinder
Rock	98.18182	1.81818
Metallic Cylinder	0	100

These results demonstrated that in both the breadth and depth first searches the improvement measure obtained the best classification, with the depth first method slightly out performing the breadth first method for both correlation and improvement measure. If these results are then compared with those obtained using Naïve Bayes, as presented in table 5, then it can be seen that apart form the correlation method, using bread first search gave an increase in classification accuracy of a maximum of 12.5%.

Table 5: Naïve Bayes Results

	Predicted Accuracy
Test	80.76923
Training	82.69231

Test Confusion Tableau:

Predicted Class/True Class	Rock	Metallic Cylinder
Rock	88.09524	11.90476
Metallic Cylinder	24.19355	75.80645

Training Confusion Tableau:

Predicted Class/ True Class	Rock	Metallic Cylinder
Rock	85.45455	14.54545
Metallic Cylinder	20.40816	79.59184

These results also highlight the trade off between granularity and dimensionality as good classification results are observed using only two labels on the 60 attributes.

These results can be comparable with Gorman and Sejnowski [9] who experiment with a back propagation neural network with 60 inputs and up to 24 hidden nodes, which are illustrated in table 6 and those of Frieβ, Cristianini and Campbell [5] who used a Kernel Adatron Algorithm obtaining a classification accuracy of 95.2%.

Table 6: Gorman and Sejnowski

Hidden Nodes	0	2	3	6	12	24
Accuracy on Test Set %	73.1	85.7	87.6	89.3	90.4	89.2

6.3 Glass Identification Database

This problem is taken from the UCI online repository [14] and is constructed by forensic scientists. The database contains 214 examples of 7 different types of glass fragments found at the scenes of crime, which if correctly identified can be used as evidence. The attributes supplied are as follows:

1	*RI: Refractive index*
2	*Na: Sodium (unit measurement: weight percent in corresponding oxide, as are attributes 4-10)*
3	*Mg: Magnesium*
4	*Al: Aluminium*
5	*Si: Silicon*
6	*K: Potassium*
7	*Ca: Calcium*
8	*Ba: Barium*
9	*Fe: Iron*

Type of glass:

1	Building windows float processed
2	Building windows non-float processed
3	Vehicle windows float processed
4	Vehicle windows non-float processed (no examples)
5	Containers
6	Tableware
7	Headlamps

Here, splitting each sub-class evenly in the problem produced a test set of 109 examples and training set of 105 examples. Both search and grouping methods were applied, with each attribute having 3 labels place over their domain in a non-uniform manner and with attributes 8 and 9 being discarded. The experiment was repeated 100 times with randomly generated data sets constructed in the same manner and the following average classification accuracy obtained:

Breath First Search Results:

Table 7: Correlation Measure Results

	Average	Upper Bound	Lower Bound	Variance	Uncertain
Test	65.33945	76.14679	55.9633	4.84518	1.76147
Training	87.66667	95.2381	80.95238	3.12223	0

Threshold = 0.005, **Max Grouping** = 4

Table 8: Improvement Measure Results

	Average	Upper Bound	Lower Bound	Variance	Uncertain
Test	67.11927	77.98165	59.63303	4.40887	3.51376
Training	91.60952	98.09524	86.66667	2.39309	0

Threshold = 0.895, **Max Grouping** = 4

Depth First Search Results:

Table 9: Correlation Measure Results

	Average	Upper Bound	Lower Bound	Variance	Uncertain
Test	65.75229	77.06422	54.12844	4.7371	2.87156
Training	90.09524	95.2381	83.80952	2.62208	0

Threshold = 0.005, **Max Grouping** = 4

Table 10: Improvement Measure Results

	Average	Upper Bound	Lower Bound	Variance	Uncertain
Test	66.24771	75.22936	57.79817	4.22675	4.00917
Training	91.50476	97.14286	86.66667	2.18293	0

Threshold = 0.895, **Max Grouping** = 4

These results show that the breadth first search slightly outperformed the depth first search in both the correlation and improvement measure even thought the differences are not significant. Further we can make a comparison with the standard Naïve Bayes classifier, the results of which are presented in table 11. These results illustrate that now the breadth first search is obtaining the highest classification accuracy with an improvement from Naïve Bayes of 3.3%.

Table 11: Naïve Bayes Results

	Average	Upper Bound	Lower Bound	Variance	Uncertain
Test	63.78899	75.22936	52.29358	4.86404	0
Training	76.60952	83.80952	68.57143	3.04559	0

Using this database a direct comparison can be made with results from Baldwin, Lawry and Martin (see [2]), who split the database into two equal training and test sets of 107 examples. Here using a breadth search with the improvement measure and setting the threshold to 0.895 with a max grouping of 4 attributes a classification accuracy of 71.03% can be obtained on the test set and 92.52% on the training set. This compares well with the results of 71% on the test set using a mass assignment prototype method [2] and a test set accuracy of 68% using a mass assignment ID3 system [3].

7 Conclusion

This paper shows that it is possible to obtain good classification accuracy using the Semi-Naïve Bayesian framework presented. These results also highlight a theorem due to Wolpert and Macready entitled "*No Free Lunch Theorems For Search*" [13], who suggested that no search can in general obtain optimum classification result for all problems. This is apparent if a comparison is made between the sonar data results section 6.2, were the depth first search method given the best classification accuracy, against the glass database, section 6.3, were the best classification result are obtained by using a breadth first search.

It has also been demonstrated that correlation is not always a good discriminator between classes. A better approach is to directly measure the improvement in discrimination obtained by any particular grouping of variables, whilst balancing granularity and dimensionality in the problem.

The methods described here represent an ongoing development of the proposed framework and further work is needed to optimise classification performance. Furthermore, the potential power of using "*natural language querying*" in such a framework has still to be investigated.

References

[1] J.F. Baldwin, T.P. Martin, B.W. Pilsworth (1995), *FRIL – Fuzzy and Evidential Reasoning in Artificial Intelligence*, Research Studies Press Ltd, John Wiley.

[2] J.F. Baldwin, J. Lawry, T.P. Martin, "A Mass Assignment Method for Prototyping Induction", *International Journal of Intelligent Systems*, Vol. 14, No.10, 1999, pp 1041-1070.

[3] J.F. Baldwin, J. Lawry, T.P. Martin, "Mass Assignment Based Induction of Decision Trees on Words", *Proceedings of IPMU98*, Paris, France, 1998.

[4] R.E. Bellman (1961), *Adaptive Signal Control Processing*, Princeton University Press, Princeton NJ.

[5] Th. Frieß, N. Cristianini, C. Campbell, "The Kernel-Adatron: a Fast and Simple Learning Procedure for Support Vector Machines", *Proceedings of the Fifteenth International Conference on Machine Learning,* 1998, pp 188-196.

[6] I. Kononenko, "Semi-naive Bayesian classifier", *EWSL-91, Porto, Springer*, 1991, pp 206-219.

[7] D.D. Lewis, "Naïve (Bayes) at Forty: The Independence Assumption in Information Retrieval", *Machine Learning ECML-98*, LNAI 1398, 1998 pp 4-15.

[8] B.R. Gaines, "Fuzzy and Probabilistic Uncertainty Logic", *Journal of Information and Control 38*, 1978, pp 154-169.

[9] R. P. Gorman and T. J. Sejnowski, "Analysis of Hidden Units in a Layered Network Trained to Classify Sonar Targets", *Neural Networks*, 1988, Vol.1, pp 75-89.

[10] J. Lawry, "A Voting Mechanism for Fuzzy Logic", *The International Journal of Approximation Reasoning 19*, 1998, pp 315-333.

[11] J. Lawry, "Label Prototypes for Modelling with Words", *Proceedings of NAFIPS*, 2001.

[12] J. Lawry, "Label Semantics: A Formal Framework for Modelling With Words", *Symbolic and Quantitative Approaches to Reasoning With Uncertaint*y, Lecture Notes in Artificial Intelligence, Springer-Verlag, 2001, pp 374-384.

[13] D.H Wolpert, W.G. Macready, "No Free Lunch Theorems for Search", *Technical Report SFI—TR—95—02—010*, Santa Fe Institute, 1995.

[14] WWW, UCI Machine Learning Repository http://www.ics.uci.edu/~mlearn/MLRepository.html.

[15] L.A Zadeh (1996), "Fuzzy Logic = Computing With Words", *IEEE Transaction on Fuzzy Systems*, Vol.4, No.2, pp 103-111.